Fall, 2023 Week 3 2023.09.25

組別:_____ 簽名:_____

[group1]

- 1. Choose true statements.
 - A. Both register and memory are built directly in the processor.
 - B. In MIPS, words must start at addresses that are multiples of 4.
 - C. We can move the data from one register to another register by using add instruction.
 - D. For subtraction in MIPS, you would use the "subi" instruction with a negative immediate value. For example, subi \$t0, \$t1, 5 # \$t0 = \$t1 5.
 - E. Given the following C code:A[9] = h + A[8]

h in \$s2, base address of A in \$s3

This is compiled MIPS code: lw \$t0, 32(\$s3) add \$t0, \$s2, \$t0 sw \$t0, 36(\$s3)

Ans:

- A. False. Only the register is in the processor.
- B. True. In the MIPS architecture, instructions and data words are typically aligned to addresses that are multiples of 4 bytes. The alignment is required to ensure proper memory access and avoid unaligned access issues.
- C. True. In MIPS, you can move data from one register to another by using the "add" instruction with the source and destination registers being the same. For example, add \$t0, \$t1, \$zero would move the contents of \$t1 to \$t0.
- D. False. In MIPS, subtraction can be performed by using the "addi" instruction with a negative immediate value. The reason for not having the subtract immediate instruction is that it reduces the number of instructions needed, which in turn reduces the complexity of the hardware.
- E. True.

[group8]

2. Please translate the following MIPS instructions to C code:

variable	f	а	b	base address of C	d	е	g	
register	\$s0	\$s1	\$s2	\$s3	\$s4	\$s5	\$s6	
w \$t0_20(\$c2)								

lw \$t0, 20(\$s3)

add \$s1, \$s2, \$t0 addi \$s4, \$s5, 7

lw \$t1,12(\$s3) addi \$s6, \$t1, -2 add \$t2, \$s1, \$s4

add \$s0, \$s6, \$t2

Ans:

a = b + C[5]; d = e + 7; g = C[3] - 2; f = a + d+ g;

[group5]

3. Given a memory table M, and assume \$s0 = 1008.

lw \$t0, -8(\$s0)	Men	nory
lw \$t1, 8(\$s0)		
add \$t2, \$t0, \$t1	1000	<u>.</u>
sw \$t2, 0(\$s0)	1004	0
lw \$t3, -4(\$s0)	1008	91
lw \$t4, 4(\$s0)	1012 2	5
sub \$t5, \$t3, \$t4	1016	7
sw \$t5, 8(\$s0)		

After the following instructions are executed, which segments of the memory are modified? What are the values of those modified segments?

Ans: lw \$t0, -8(\$s0) (\$t0 = 234) lw \$t1, 8(\$s0) (\$t1 = 177) add \$t2, \$t0, \$t1 (\$t2 = 234 + 177 = 411) sw \$t2, 0(\$s0) (M[1008] = 411) lw \$t3, -4(\$s0) (\$t3 = 110)
lw \$t4, 4(\$s0) (\$t4 = 25)
sub \$t5, \$t3, \$t4 (\$t5 = 110 - 25 = 85)
sw \$t5, 8(\$s0) (M[1016] = 85)

the values of M[1008] and M[1016] changed. M[1008] = 411 M[1016] = 85

[group6]

- 4. Please justify the following statement (T/F).
- (A) Immediate operands are values that are directly encoded within the instruction itself.
- (B) Immediate operands can always improve the performance of instructions with constant.
- (C) We can use constant zero to move between registers.

Ans:

(A) True.

(B) False, if the constant size is larger than available operand size, we may need to split the constant into smaller parts or use multiple instructions to load the constant, which can be less efficient than accessing the constant from memory or using a register.

(C) True.

[group4]

5. Assume a 32-bit machine. Assume each of the following questions starts from the table values; that is, DO NOT use value changes from one question as propagating into future parts of the question.

Register	Value	Memory Location	Value
R1	12	12	16
R2	16	16	20
R3	20	20	24
R4	24	24	28

- a. Give the values of R1, R2 after this instruction: addi R2, R1, 5
- What values will be in R1 and R3 after this instruction is executed: lw R3, 12(R1)

Ans:

a. **R2=17, R1=12**

R2=12+5 R1 remains unchanged

b. **R3=28, R1=12**

Register R1 contains 12 12+12=24 Memory location 24 contains 28, so R3=28 R1 remains unchanged

[group12]

6. Please choose the correct answer(s).

(a) Unlike high-level languages like C programming, assembly don't use variables.

(b) Registers can improve code density, but it may make memory traffic worse.

(c) Computer Architecture = ISA (Instruction Set Architecture) + Compiler

(d) Syntax of basic MIPS arithmetic/logic instruction is free, so that we can decide the number of operators and operands.

Ans:

(a)

- (b) Registers can improve code density and reduce memory traffic.
- (c) Computer Architecture = Instruction Set Architecture + Machine Organization.
- (d) Syntax of basic MIPS arithmetic/logic instruction is rigid: 1 operator, 3 operands (sources*2 + destination*1).

[group13]

7. 綜合題:

- a. Write one benefit of using registers.
- b. In MIPS assembly, registers like \$t0 and \$s1 are used for temporary data storage during program execution. (true or false)
- c. What is the result of "addi \$0, \$0, 7"?
- d. How to use add operator to move \$t1 to \$t2. Write an assembly code.

Ans:

- a. Registers provide a benefit as they serve as a convenient location for a compiler to store instructions temporarily.
- b. true
- c. 0, because it will do nothing
- d. add \$t2, \$t1, \$zero